Context-aware patient monitoring through sensor streams

Pieter Bonte, Filip De Turck, Femke Ongenae

Ghent University - imec, Ghent , Belgium
{pieters.bonte,filip.deturck,femke.ongenae}@ugent.be

Abstract Personalized eCare in hospitals allows to reduce cost and increase the patient satisfaction. In this demonstrator, we present a cascading reasoning platform that is able to combine the volatile sensor streams that monitor the patient with expressive reasoning to enable complex decision making, providing tailored care for each patient.

Keywords: Stream Processing, Patient Monitoring, Cascading Reasoning

1 Introduction

Hospitals struggle to reduce cost due to the financial pressure on the healthcare system while trying to deliver more personalized care to its patient. In personalized care, each individual is seen as unique and the care should be adapted for each patient individually [4]. For example, there is no fixed threshold to determine the presence of a certain diagnosis, but the threshold is based on the individual's personal characteristics. Personalized care through personalized monitoring also allows to intervene in time-critical situations, e.g. notifying the closest nurse with the right capabilities to aid a patient who is having a seizure, as quickly as possible.

To enable personalized care in hospitals through sensor monitoring, sensor streams need to be combined with background knowledge, allowing to determine the exact thresholds to monitor for each patient. The background knowledge describes the medical knowledge and relation between symptoms & pathologies, the pathology of the patient, the sensitivity of each condition to different stimuli, the capabilities of the nurse, etc. To make the right decisions, advanced logic, such as Description Logic (DL) reasoning, are necessary. These reasoning techniques allow to take the domain knowledge into account and infer implicit facts about the data, e.g. which sensor readings should be taken into account. However, the complexity of the expressive reasoning technique is often too high to be executed directly over high-velocity streams of sensor data. Therefore, we propose a cascading reasoning platform, combining RDF Stream Processing (RSP) to filter only the relevant parts from the sensor streams with expressive DL reasoning to infer implicit statements in the data and Event Processing (EP) to detect temporal patterns.

We focus on a use case where we monitor characteristics of the room in function of the light and sound levels and the movement of the patient. Figure 1 a) shows the set up of the use case. This is respectively done with light, sound and movement sensors integrated in the room. The personnel can be localized and notified through the use of

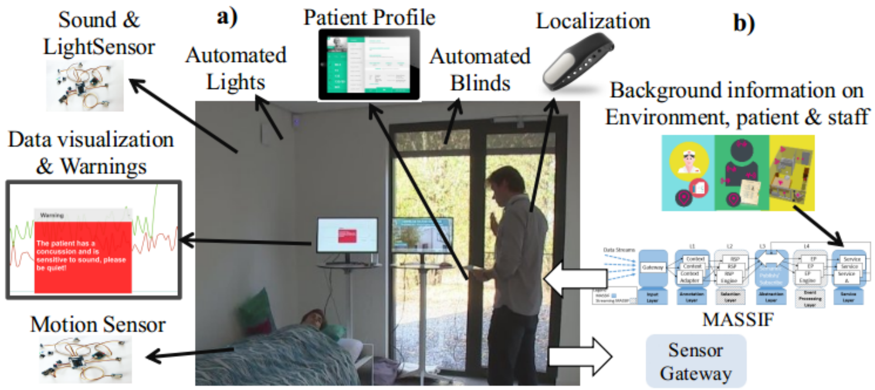

Figure 1. Overview of the use case and how the various components interact.

smart wearables. Based on the pathology of the patient, e.g. concussion or photosensitive epilepsy, the patient's sensitivity to the sound and/or light inside the room is determined.

Therefore, when the patient is diagnosed with one of these diseases, the room will react to these stimuli, e.g. when the light levels are too high, the lights will automatically dim and the blinds shut. When the sound levels are too high, the television mutes, the visitors are requested to stay quiet or a nurse is notified to investigate the situation. In case of epilepsy, when the patient is having a seizure, detected through the movement sensors, the personnel is automatically alarmed. Note that only those sensor streams are taken into account that are relevant to the pathology of the patient.

2 Architecture

To combine the data streams with expressive reasoning techniques, we built on the idea of cascading reasoning [5], which states that expressive reasoning over high-velocity streams can be achieved in a layered approach. Each layer minimizes the data change frequency, by selecting only those parts of the data that are relevant for further processing and increases the complexity of processing.

We extended the MASSIF platform [1] to enable cascading reasoning by adding two layers: a Selection Layer and an Event Processing Layer. The architecture of the extended platform is visualized in Figure 2. We explain each of these layers in detail:

- Input Layer: Serves as an entry point to the platform.
- Annotation Layer: An optional layer that allows to annotate raw data to the semantic model through the use of RML[1].
- Selection Layer: Selects those parts of the RDF stream that are relevant for further processing.
- Abstraction Layer: Receives the selected triples from the *Selection Layer* and abstracts them to high-level events.
- Event Processing Layer: This layer can perform event processing on the abstractions from the previous layer. For example, temporal dependencies between abstractions can be detected in this layer.

[1] http://rml.io

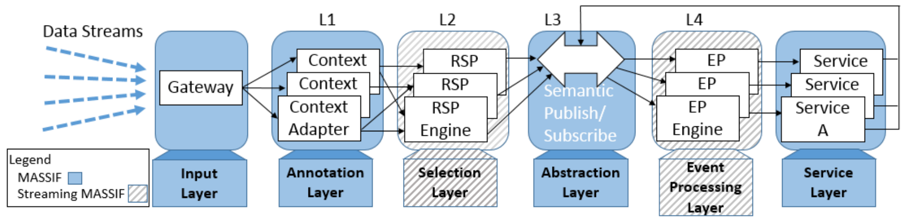

Figure 2. The Cascading Reasoning extension of the MASSIF platform.

– Service Layer: Allows various services to subscribe to the abstractions or to the results of the *Event Processing Layer*. Each service can perform more advanced reasoning and processing on the received data.

3 Implementation & Demonstrator

In this section we elaborate on how we implemented the concussion/epilepsy use case on the extension of the MASSIF platform, which will be shown in the demonstrator. To capture the data streams, we utilize the DYAMAND sensor gateway [3] that allows plugins to interact with sensors and actuators. The captured sensor readings are directly mapped to the SSN ontology [2] in one of these plugins, allowing the sensor gateway to stream out semantic annotated data. Figure 3 visualizes a part of the used ontology that describes the relation between the diagnosis's and the sensor observations. We will now detail how each of the layers of the MASSIF platform were instantiated:

– *Selection Layer:* The ontological background data describes for each pathology the sensitivity to each kind of stimuli. For example, patients with a concussion should not be exposed to sound levels above 40dB. When a patient is diagnosed with a concussion, the profile of the patient is updated and the selection layer will now select the sound observations above 40dB. Similar situations occur for the light and movement observations. The demo will show that the sensor readings are only taken into account when the diagnosis has been fixed. The selection layer thus consists of an RSP engine that specifically filters the data based on pathology of the patient.
– *Abstraction Layer:* This layer will abstract the selected data to high-level concepts. In the case of the concussion, the high sound level will be abstracted as a SoundThresholdObservation, which is described in the background knowledge (and is a subclass of ThresholdObservation).
– *Event Processing Layer:* This layer allows to detect temporal dependencies between events. For example, when there is a ThresholdObservation but there is no observation of a staff member being present in the room for a period of 3minutes, an additional nurse should be notified.
– *Service Layer:* One service receives all the threshold observations, localization of the staff, diagnosis of the patients, etc. When a threshold observation is detected, the service decides how to interact with the room. When the light/sound levels are exceeded, the room reacts accordingly, the blinds shut, the lights go out, the tv turns off/mutes, a notification is shown on the tv to ask the visitors to be quiet or a staff

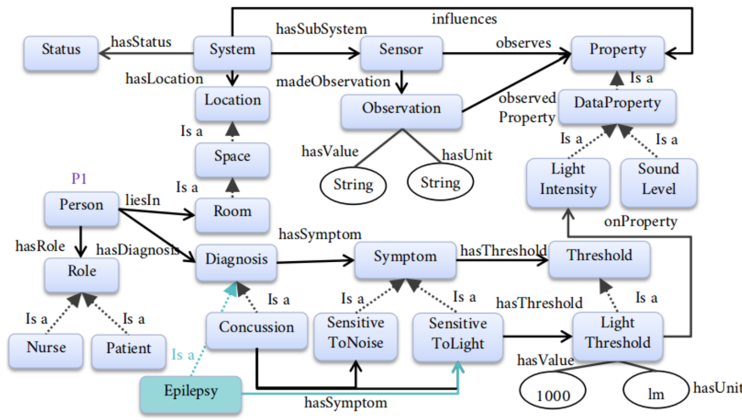

Figure 3. Visualization of the used ontology.

member is notified. The service will generate *actions* that describe its decisions. These *actions* are picked up by a second service that will interact with the sensor gateway and tell the room how to steer is actuators to make the life of the patients as pleasant as possible.

We did not take the annotation layer into account since the sensor gateway is already producing annotated data. Figure 1 visualizes the different components in the demonstrator. The demo also shows the extensibility of the platform, easily incorporating another diagnosis, i.e. photosensitive epilepsy, where the patient's movement needs to be monitored in order to detected seizures[2]. Due to portability issues, the blinds will be virtualized.

4 Conclusion

In this paper, we presented a flexible patient monitoring system that is able to reason over sensor data streams by building upon the principles of cascading reasoning. We propose a layered platform consisting of 1) an RSP layer that selects those parts of the sensor data that are relevant for further processing, 2) an abstraction layer that is able to perform expressive reasoning to infer implicit facts in the selected data and 3) an event processing layer that can detect temporal patterns within the data.

References

1. Bonte, P., et al.: The MASSIF Platform: a Modular & Semantic Platform for the Development of Flexible IoT Services . Knowledge and Information Systems pp. 1–38
2. Compton, M., et al.: The ssn ontology of the w3c semantic sensor network incubator group. Web semantics: science, services and agents on the World Wide Web 17, 25–32 (2012)
3. Nelis, J., et al.: Dyamand: dynamic, adaptive management of networks and devices. In: Local Computer Networks (LCN), 2012 IEEE 37th Conference on. pp. 192–195. IEEE (2012)
4. Snyderman, R.: Personalized health care: from theory to practice. Biotechnology journal 7(8), 973–979 (2012)
5. Stuckenschmidt, H., et al.: Towards Expressive Stream Reasoning (2010)

[2] A video can be found on `http://pbonte.github.io/ISWC2018Demo/`